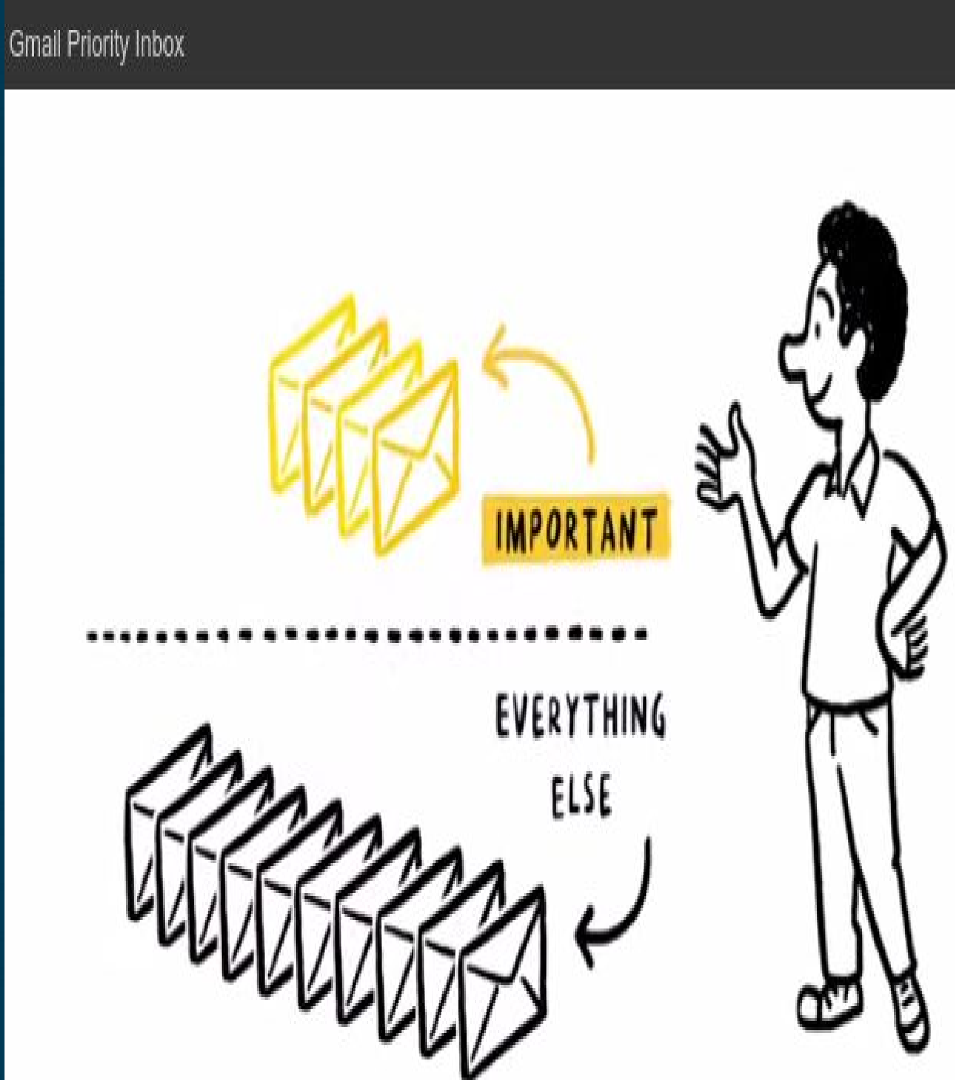
Malware Analysis using machine learning

Presented by Utsava Verma & Mohit Sharma

Outline

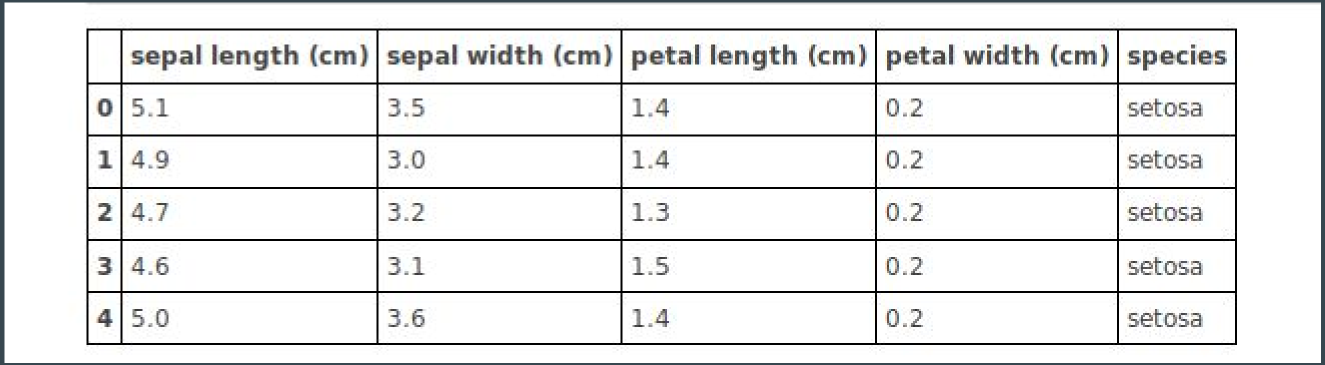
- What is machine learning
- ML models
- Why malware analysis
- Methods of malware analysis -static and dynamic
- PE file format
- Our strategy
- Future works

MACHINE
LEARNING


Gmail Priority Inbox
IMPORTANT
EVERYTHING
ELSE

What is Machine Learning

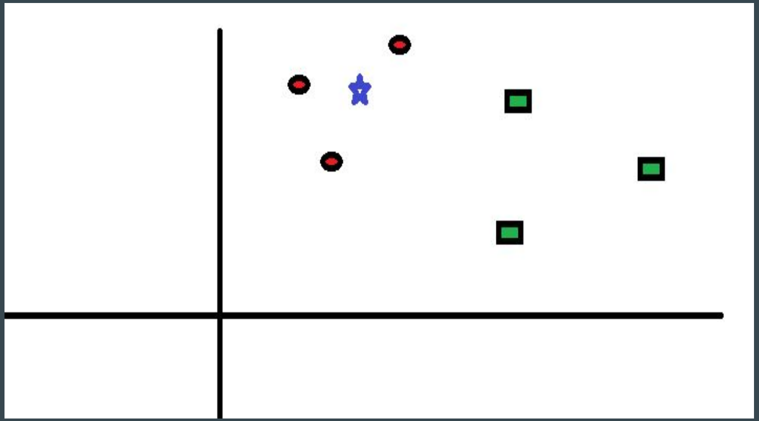
Machine learning is a method of data analysis that automates analytical model building.

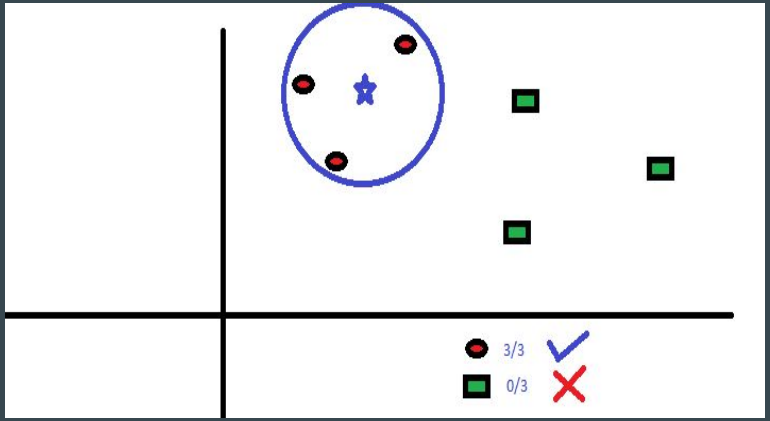


	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

K-Nearest Neighbors

- Pick a value for k.
- Search for the k observations that are nearest to the unknown iris.
- Use the most popular response value from the k nearest neighbors as the predicted response.





k=3

Logistic Regression

- A classification algorithm.

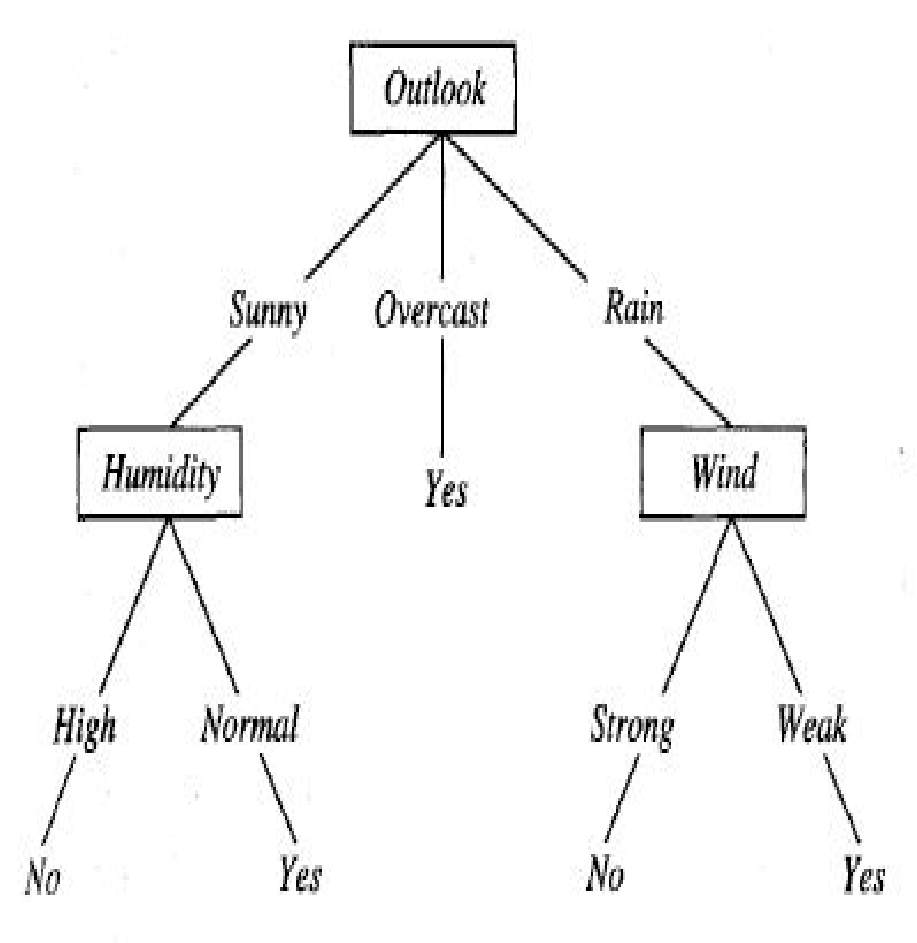
- Used to predict a binary outcome (1/0) given a set of independent variables.

- Eg -To determine the likelihood of a patient's successful response to a medical treatment. Input variables - age,weight,blood pressure and cholesterol levels.

Decision tree & Random forest

- To create a model that predicts the value of a target variable based on several input variables.

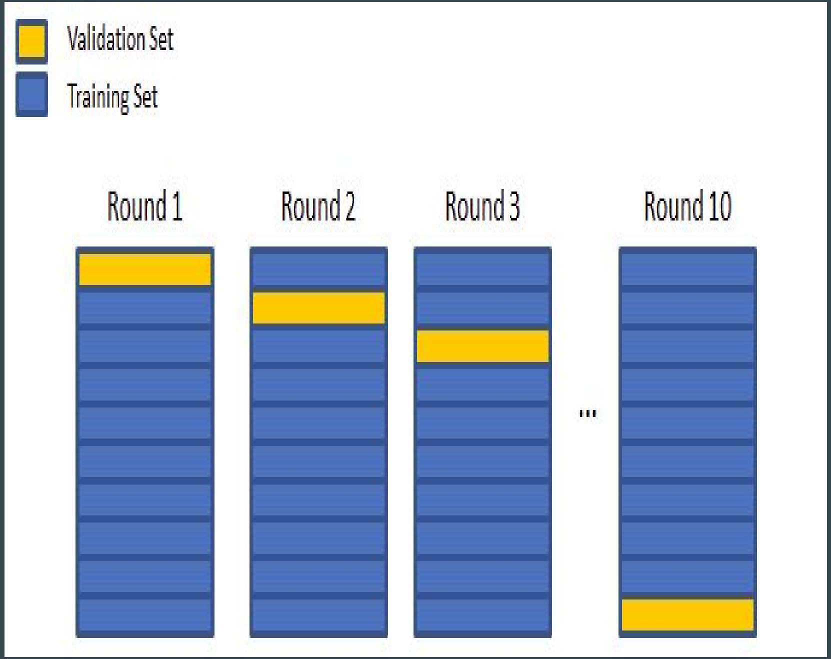
- Random forests operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification).



K-Fold cross validation

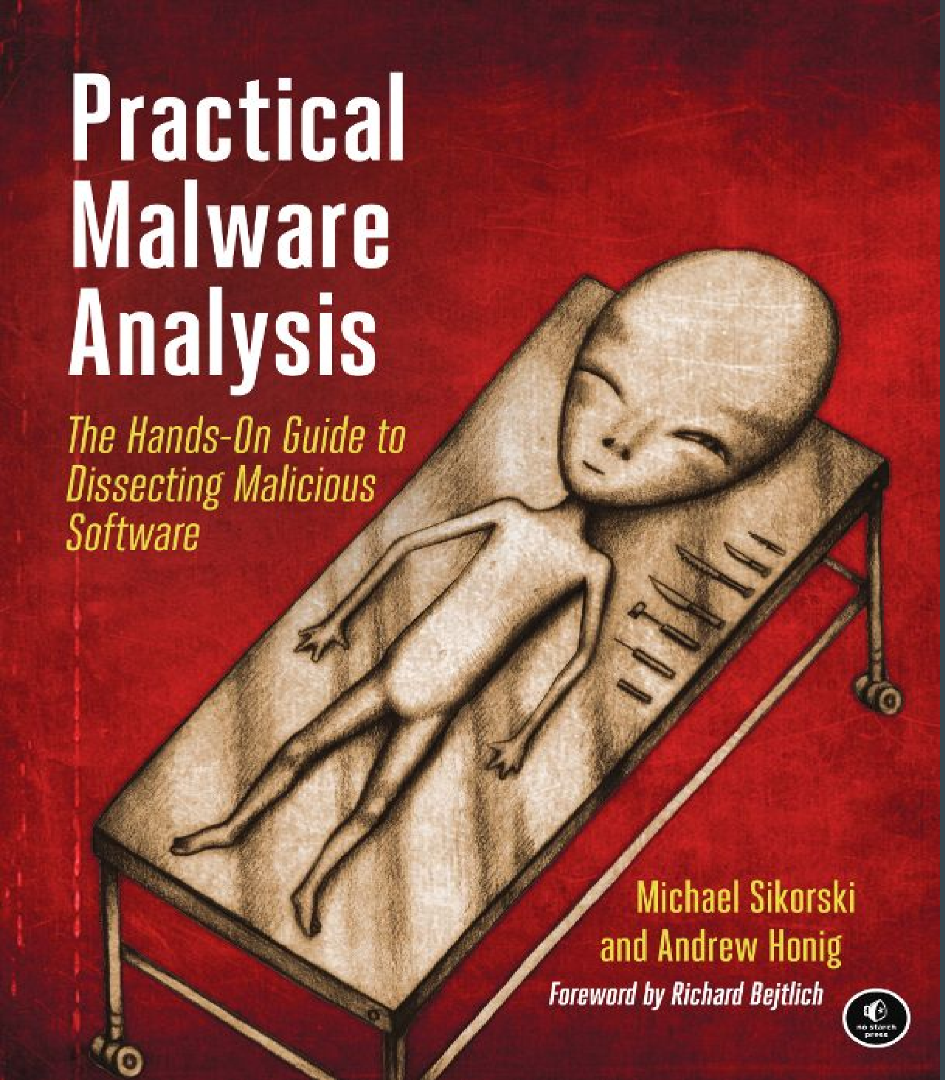
- A technique to evaluate predictive models by partitioning the original sample into a training set to train the model, and a test set to evaluate it

- Repeated k times.



Problem Statement

Apply ML models to classify a sample as benign or malicious instead of using just hashes.



Malware analysis

- What is it?
- Why do we need it?
- That’s so cool! How do I start??

Static Analysis vs Dynamic Analysis

Static Analysis	Dynamic Analysis
Examines the executable files without viewing the actual instructions	Observing the behavior of the malware while it is actually running on a system

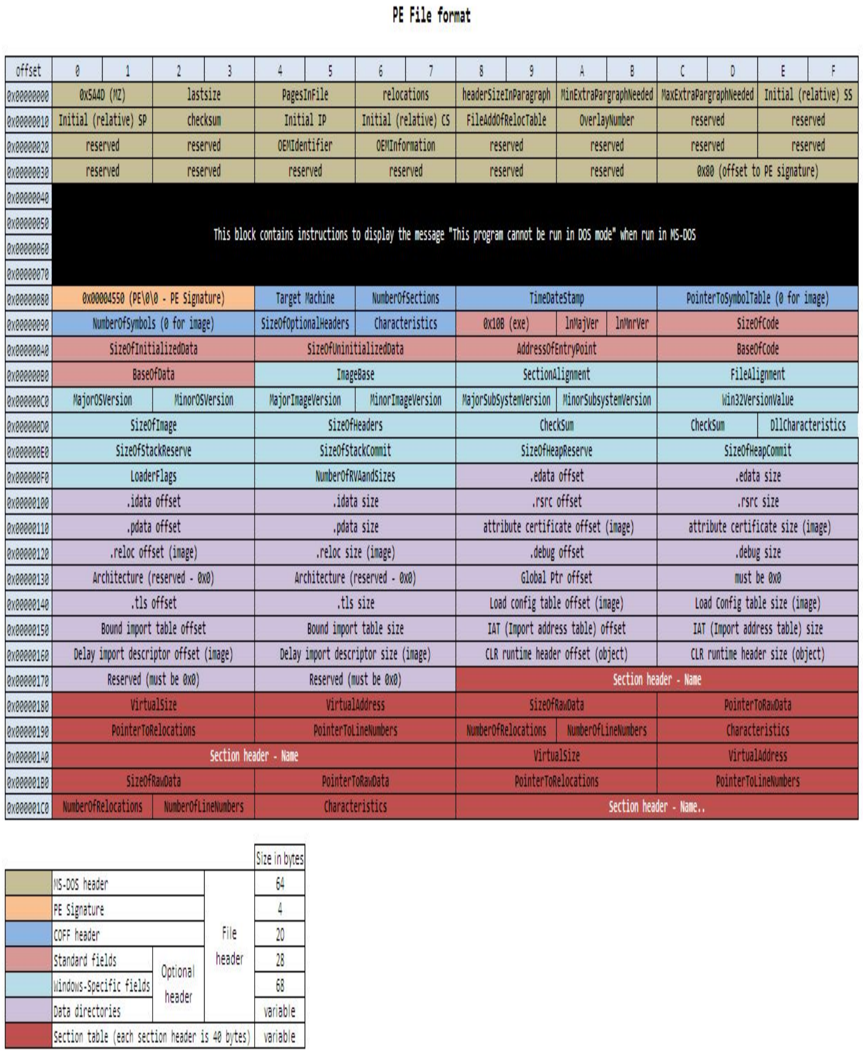
Our Strategy - Static Analysis

- Let’s Use Machine Learning
- Features matrix
- Target Vector
- Use a model - Train & Predict

- Okay Cool! Let’s get started!

PE File Structure

- That’s so much data!
- How do I pick the correct features??



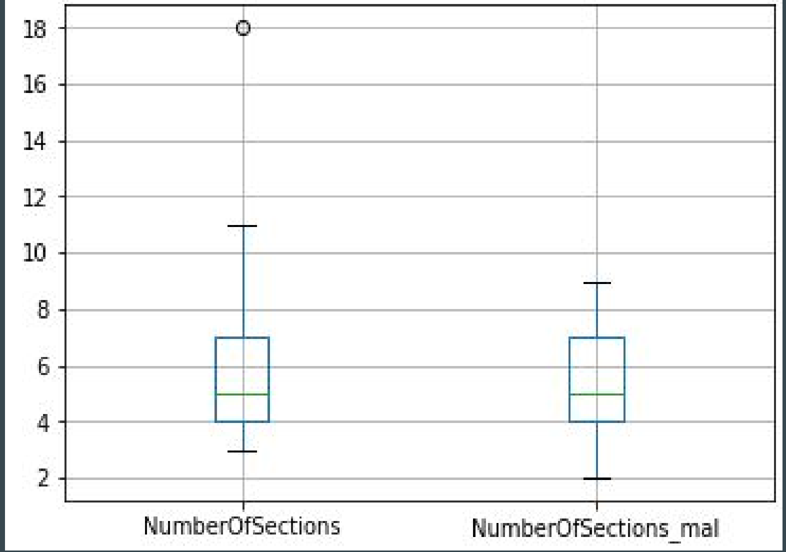
PE File format

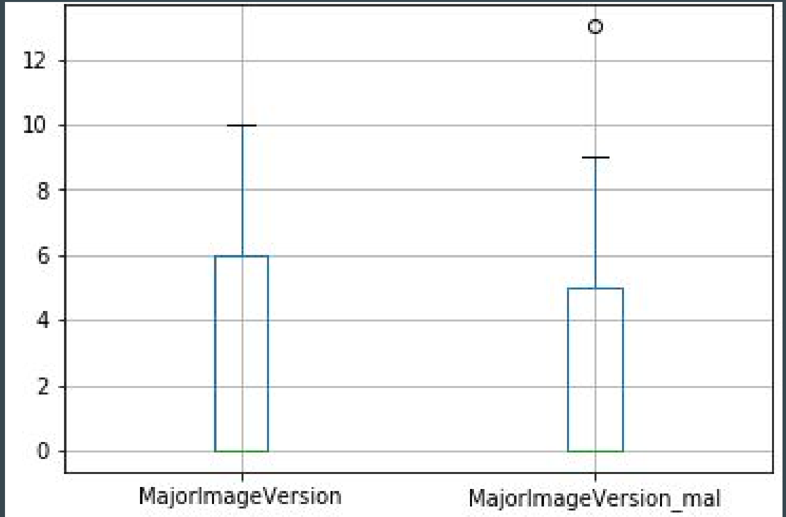
offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x00000000	0x5A4D (MZ)		lastsize		PagesInFile		relocations		headerSizeInParagraph		MinExtraPargraphNeeded		MaxExtraPargraphNeeded		Initial (relative) SS	
0x00000010	Initial (relative) SP		checksum		Initial IP		Initial (relative) CS		FileAddOfRelocTable		OverlayNumber		reserved		reserved	
0x00000020	reserved		reserved		OEMIdentifier		OEMInformation		reserved		reserved		reserved		reserved	
0x00000030	reserved		reserved		reserved		reserved		reserved		reserved		0x80 (offset to PE signature)			
0x00000040	This block contains instructions to display the message "This program cannot be run in DOS mode" when run in MS-DOS															
0x00000050																
0x00000060																
0x00000070																
0x00000080	0x00004550 (PE\0\0 - PE Signature)				Target Machine		NumberOfSections		TimeDateStamp				PointerToSymbolTable (0 for image)			
0x00000090	NumberOfSymbols (0 for image)				SizeOfOptionalHeaders		Characteristics		0x10B (exe)		lnMajVer	lnMnrVer	SizeOfCode			
0x000000A0	SizeOfInitializedData				SizeOfUninitializedData				AddressOfEntryPoint				BaseOfCode			
0x000000B0	BaseOfData				ImageBase				SectionAlignment				FileAlignment			
0x000000C0	MajorOSVersion		MinorOSVersion		MajorImageVersion		MinorImageVersion		MajorSubSystemVersion		MinorSubsystemVersion		Win32VersionValue			
0x000000D0	SizeOfImage				SizeOfHeaders				CheckSum				CheckSum		DllCharacteristics	
0x000000E0	SizeOfStackReserve				SizeOfStackCommit				SizeOfHeapReserve				SizeOfHeapCommit			
0x000000F0	LoaderFlags				NumberOfRVAandSizes				.edata offset				.edata size			
0x00000100	.idata offset				.idata size				.rsrc offset				.rsrc size			
0x00000110	.pdata offset				.pdata size				attribute certificate offset (image)				attribute certificate size (image)			
0x00000120	.reloc offset (image)				.reloc size (image)				.debug offset				.debug size			
0x00000130	Architecture (reserved - 0x0)				Architecture (reserved - 0x0)				Global Ptr offset				must be 0x0			
0x00000140	.tls offset				.tls size				Load config table offset (image)				Load Config table size (image)			
0x00000150	Bound import table offset				Bound import table size				IAT (Import address table) offset				IAT (Import address table) size			
0x00000160	Delay import descriptor offset (image)				Delay import descriptor size (image)				CLR runtime header offset (object)				CLR runtime header size (object)			
0x00000170	Reserved (must be 0x0)				Reserved (must be 0x0)				Section header - Name							
0x00000180	VirtualSize				VirtualAddress				SizeOfRawData				PointerToRawData			
0x00000190	PointerToRelocations				PointerToLineNumbers				NumberOfRelocations		NumberOfLineNumbers		Characteristics			
0x000001A0	Section header - Name								VirtualSize				VirtualAddress			
0x000001B0	SizeOfRawData				PointerToRawData				PointerToRelocations				PointerToLineNumbers			
0x000001C0	NumberOfRelocations		NumberOfLineNumbers		Characteristics				Section header - Name..							

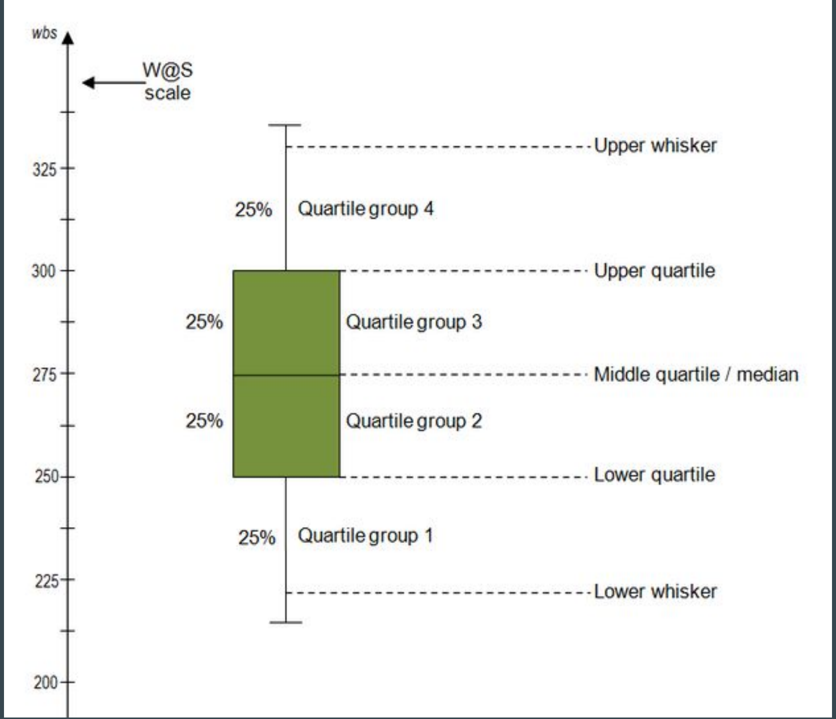
				Size in bytes
🟫	MS-DOS header		File header	64
🟧	PE Signature			4
🟦	COFF header			20
🟥	Standard fields	Optional header		28
🟦	Windows-Specific fields			68
🟪	Data directories			variable
🟥	Section table (each section header is 40 bytes)			variable

Boxplots

- What are boxplots?
- Why do we need them?







“Sweetheart, my neural net predicts that you and I are 98.9% compatible. Will you be my Valentine?”

ML Models

- Comparison Table

	Logistic Regression	KNN	Decision Tree	Random Forest
Accuracy	80	94	99.98	96

So What did we learn ?

- Practical Machine Learning
- Static Analysis
- Existence of CTF’s

Current Status - Single ;)

- Working Product
- http://localhost:4555

Future Work

- Explore Neural Networks
- Continue working on our product
- Dynamic Analysis

Acknowledgement

- Prof. Sandeep Shukla
- Pranjul Ahuja & Vineet Purswani
- Colleagues

Thank You